Predicting User Influence under the Environment of Big Data

Jun Zhou[1,2,3], Guiping Wu[3], Manshu Tu[3], Bing Wang[3], Yan Zhang[3], Yonghong Yan[2,3,4]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]The Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, Chinese Academy of Sciences, Beijing, China
[4]Xinjiang Laboratory of Minority Speech and Language Information Processing, Xinjiang Technical Institute of Physics and Chemistry, Chinese Academy of Sciences, Ürümqi, China
e-mail: zhoujun@iie.ac.cn, {wuguiping, tumanshu, wangbing, zhangyan, yanyonghong}@hccl.ioa.ac.cn

***Abstract*—A person is considered as an influential individual when his behaviors can trigger other people's reactions. Such phenomenon is called user influence in social networks. Measuring user influence provides insights into dynamics of social network interactions. This makes a fundamental step for constructing marketing strategy, recommendation systems and so on. There have been various studies focusing on this problem. However, it lacks of a satisfactory method to measure user influence in a reasonable way. It is also worth studying on user properties and past activities that contribute to the influence of each user. In this paper, we investigate the attributes of millions of social network users and the content of their messages in order to better predict user influence. These users and messages are from Sina Weibo, which is one of the most popular social networks in China. Our first contribution is to quantify the influence of individuals within a period of time by using a new approach and find the individual influence of most users changing over time, but most changes are not significant. Our second contribution is to propose a phrase merging algorithm for obtaining high-quality phrases, which are very helpful for extracting the topics that each user is interested in. Our third contribution is to predict the influence of each user with a higher precision.**

Keywords-user influence; social network; big data; spark.

I. INTRODUCTION

As a social media platform [1], Sina Weibo allows people to post brief messages to air their opinions and their followers can forward or comment on these posts. This platform makes opinions spread faster than traditional media. Thus, it has gained huge popularity in China [2]. Due to the data richness and public availability of the microblogging system, researchers have been interested in studying user influence in social networks recently. Studying the question has a realistic significance in our real life. For example, it helps marketing effort to target on most influential individuals for delivering ads.

Some work has been made on the problem of measuring user influence. In Cha's study [3], user influence in social media was computed by using the number of followers, retweets, and mentions gained on Twitter. Another similar metric proposed by Anger et al. [4] measured user influence on account of the ratio of the number of followers and the number of friends. The underlying assumption is that if a person has more fans and pays less attention to others, he is considered to be influential. Therefore, the ratio is a little better than the former method, but it is still imprecise. To measure social networking potential of microblogging users, some other researchers have quantified user influence from the perspective of network structure. Among them, Brown et al. [5] used a modified K-shell algorithm to measure the influence of each user on Twitter. A ProfileRank algorithm for identifying influential users was proposed by Silva et al. [6]. Considering both the topical similarity between Twitter users and the network structure, Weng et al. [7] measured the influence with an extension of the PageRank algorithm. However, it is well-known that some followers purely receive messages, but may not read them, let alone they can be influenced. Therefore, these measures do not accurately capture the essence of influence by using network metrics without distinguishing followers' properties. The various approaches [8, 9, 10] currently being proposed to quantify user influence are not the obvious best choice. The work of Bakshy et al. [11, 12] has inspired our study.

Based on previous studies, we propose a new way of quantifying user influence by considering both the quantity of messages posted and their popularity. The proposed method is evaluated by using data from Sina Weibo. Based on this definition, our study reveals that the individual influence of most users changes over time, but most changes are not significant. This conclusion is in line with Akritidis's [13]. In our opinion, what is more meaningful is predicting one's influence in the future instead of ex-post analysis. We believe that user influence depends on user properties and past activities. The user properties and past activities are divided into two distinct types of features: statistical feature and topic feature. All the statistical features mentioned in our work are easy to calculate and to understand. Here we lay special emphasis on extracting topic feature by using a novel approach of phrase merging algorithm. The topic information extracted is used to improve the performance of predictive models. Thus, the influence of each user is predicted successfully.

The rest of the paper is organized as follows: a real world dataset has been prepared for our study in Section 2. In Section 3, we propose a novel approach of measuring user

978-1-5090-4499-3/17/$31.00 ©2017 IEEE

influence for everyone in Microblogs. In Section 4, we detailedly describe two types of features: statistical feature and topic feature. In Section 5, we present our experiments and analyze results. Conclusions and future work are given in Section 6.

II. SINA WEIBO DATASET

To prepare for the experiment, we collected large amounts of data from Sina Weibo over the two-year period of April 1 2013 - March 31 2015. As for a user, we can obtain his/her screen name, favorites, description, gender, registration date, number of followers and friends and so on. As for a message, we can obtain the author of the message, the content of the message, post time, whether there are pictures contained, the number of reposting it and the number of replying to it.

Since one's posts can represent his viewpoint, we restrict the study to seed content for each user, meaning the messages are not reposted from others. There are in total 1,262,518 users with their 114,286,565 seed messages in our dataset. The total number of reposting and replying to a message is regarded as the number of reacting to it, because reposting behavior and replying behavior are both responses to it. As Table I shows, the average number of publishing seed messages is 92.643 for each user in this two years. The average number of reacting to each message is 5.537 and the median is 1, implying that most messages do not attract too much attention.

TABLE I. STATISTICS OF USERS AND THEIR SEED MESSAGES

Statistics	*Number of posts per user*	*Number of reactions per post*
Minimum	3	0
Maximum	2000	1726182
Median	31	1
Mean	92.643	5.537
StdDev	202.315	1203.825

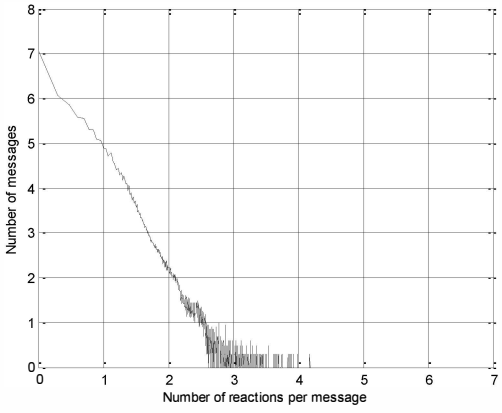

Figure 1. Distribution of posts with different reactions. Here horizontal axis represents the logarithm of the number of reactions per message and vertical axis represents the logarithm of the number of posts.

The distribution of posts with different reactions is shown in Figure 1. It follows an approximately power-law distribution, which is in line with the actual situation. Figure 1 show that only a small portion of the messages are reacted thousands of times.

III. COMPUTING USER INFLUENCE

This section introduces a new way to identify user influence. Based on this approach, we get some meaningful conclusions by observing the individual influence change of each user within a period of time.

A. A Novel Approach for Measuring User Influence

In the paper, user influence corresponds to a particular definition that one's ability to make his followers react to his messages. Measuring user influence is broken down into two steps. First, we quantify the influence score of a message. When influenced by a message, people usually forward or reply to it. The two behaviors are both the marks of the occurrence of influence. Therefore, the total number of reposting and replying to a message is taken as its influence score. Second, we refer to the definition of h-index [14] to compute the influence score for each user. In a similar way, we regard one's influence score as h who has posted h messages each of which influence score is at least h in a period of time. The higher one's score is, the greater the influence he owns. Here time is set to one year, because one year is a complete cycle and time is too short to compute the influence score of each user effectively using the definition.

As the example shows in Table II, f is the function that corresponds to the count of reactions for each post. If a user with five posts A, B, C, D, and E with 13, 8, 7, 5, and 1 reactions, respectively, his influence score is equal to 4 because the fourth post has 5 reactions and the fifth has only 1. Similarly, if the five posts have 12, 9, 3, 1, and 0, then his influence score is 3 because the fourth post has only 1 reaction.

TABLE II. A EXAMPLE OF COMPUTING USER INFLUENCE SCORE WITH OUR APPROACH

f(A)	*f(B)*	*f(C)*	*f(D)*	*f(E)*	*User influence score*
13	8	7	5	1	4
12	9	3	1	0	3

The definition is designed to improve upon simpler measures such as the number of posts or reactions. It reflects both the number of posts and the number of reactions per message. It offers a more reasonable quantitative method of determining user influence and gives us a deeper understanding the concept of user influence in social media.

B. User Influence Changes over Time

According to the definition of computing user influence, we compare the change of the first-year influence score and the second-year influence score for each user. Surprisingly, we find that 89.2% of the users have changed on individual influence. Furthermore, we want to explore how much have

changed on the whole. Therefore, the cosine similarity is used to assess the whole change. Its definition is as follows,

$$sim = 0.5 + 0.5 \times \frac{\sum_{i=1}^{n}(x_i \times y_i)}{\sqrt{\sum_{i=1}^{n}(x_i)^2} \times \sqrt{\sum_{i=1}^{n}(y_i)^2}} \quad (1)$$

where x_i, y_i are the first-year influence score and the second-year influence score respectively, and the cosine similarity value is normalized into the range [0, 1].

Since its value (sim equals 0.8015) is close to 1, we conclude that the individual influence of most users changing over time, but most changes are not significant.

IV. FEATURES

As is known to all, feature selection is a very important process that is directly related to the performance of predictive models. The related features we explore are divided into two distinct categories: statistical feature and topic feature. While statistical feature refers to user attributes that are related to the author and various statistics about the content of seed messages, topic feature is a probability distribution over several topics that each user is interested in. The detailed components of each category are introduced in the following parts.

A. Statistical Feature

The statistical features are so easy to calculate and to understand, which are referenced frequently in many research work on social media [15, 16]. For a clearer description, 19 statistical features we mention can be divided again into two parts: 13 statistical features about user properties and 6 statistical features about the content.

On the one hand, according to related work [17, 18, 19], we use the following statistics as the statistical features of user properties: number of followers, number of friends, number of microblogs, number of favorites, number of bi-followers, number of important friends, whether contains url, verification, length of screen name, length of description, date of joining, gender and the follower to friend ratio. Among these features, the number of microblogs is the number of messages posted, including seed messages and republished messages. The number of favorites is the number of other people's microblogs the user collected. Bi-followers indicate that two users follow each other mutually. Important friends are the persons who the user pays special attention to. Url is a binary variable that indicates whether there is a blog address in the personal properties. Verification means that the user has been certificated by Sina Weibo. The follower to friend ratio represents a combined statistical feature that is the ratio of the number of followers and the number of friends.

On the other hand, it takes into account the number of hashtags, links, pictures, mentions and words in the microblog and the length of the microblog as the statistical features about the content. The number of words and the length of the message are two different meanings. One indicates the number of actual words and the other is the number of characters contained. Thus, these features correspond to each message not to each user. As for each user, we regard the mean of his entire messages over a period of time as his statistical features. Based on the accessibility and effectiveness of the statistical features, researchers prefer to do some research on the content of microblogs with them [20, 21, 22]. These statistical features are also used in our follow-up experiments.

B. Topic Feature

To extract topic feature, we need to integrate the messages from each microblogging user. Since we aim to understand the topics that each user is concerned about instead of the topic that each single microblog is about, all the microblogs of each user were aggregated into a big document. Thus, each document essentially corresponded to a user.

The process of segmenting a document into 'bag-of-words' is broken down into two steps. First, the microblogs in our dataset are mainly written in Chinese. Therefore, we remove non-Chinese characters from the content, which do not help in topic modeling. Since Chinese word segmentation is different from English's, we segment Chinese words with the method Ansj [23], which is one of the most popular Chinese word-cutting methods. Thus, a document is segmented into a collection of words. Second, the frequency of each word is counted after removing stop words from the collection. In general, we set a minimum support to remove low-frequency words. Thus, a document is segmented into 'bag-of-words' that provide a new representation for documents.

1) Phrase Merging Algorithm: The main novelty in extracting topic feature is the way we transform the above original 'bag-of-words' to a high-quality 'bag-of-phrases'.

Based on statistical analysis about the occurrence of words, we consider a null hypothesis that the documents are generated from a series of independent Bernoulli trials [24]. Under this hypothesis, we provide a new quantitative measure of which two adjacent words merge the best collocation. Here all the words that would be merged must be frequent, which is the primary condition. The significance score is measured by comparing the frequency of two adjacent words with the occurrence count of each word independently. In the following definition,

$$sig(p_i, p_j) = \frac{g(p_i \& p_j)}{\min\big(g(p_i), g(p_j)\big) - g(p_i \& p_j) + 1} \quad (2)$$

where g is the function that corresponds to the actual number of a word in all the documents, p_i and p_j are two different words, and $p_i \& p_j$ indicates the consecutive co-occurrence of two words. Equation 2 computes the significance value as a robust collocation measure in selecting two adjacent words for merging. A high score stands for a high-belief that two different words are highly associated and should be merged.

On the basis of the above quantitative measure, the phrase merging algorithm we present is shown in Figure 2. It is a bottom-up iterative algorithm. At each iteration, the contiguous pair with the highest score (greater than or equal

to the threshold value we set in advance) will be selected and merged. The newly merged phrase is considered as a single unit at the next iteration. The algorithm terminates when the following merging with the highest score does not meet the threshold or when all the words have been merged into a single unit.

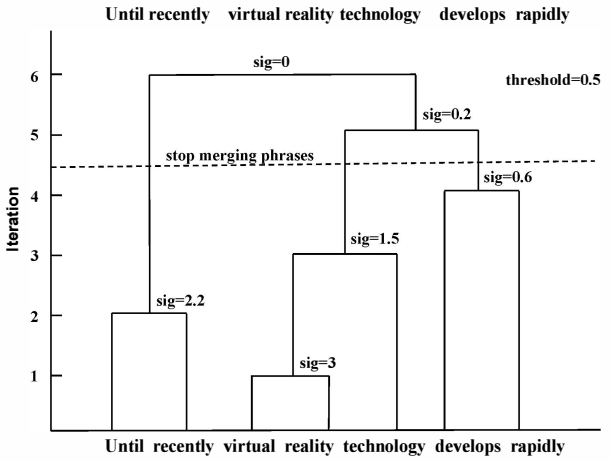

Figure 2. Phrase merging of the 'bag-of-words' of a seed message that is translated into English.

Since our phrase merging algorithm is purely data-driven, it has two main advantages. One of the advantages is that it can be implemented without knowing anything about domain knowledge and complicated language rules in advance. The other one is that the greater the amount of data is, the better the results yielded, which is suitable for big data analysis.

2) Topic Distillation: The purpose of topic distillation is to automatically identify the distribution of topics for each document i.e. each user. Latent Dirichlet Allocation (LDA) model [25, 26, 27] is used to accomplish this task.

In the paper, a big data analysis platform – Spark is applied to process the large amount of data. Spark is an offline big data processing framework and has attracted a great deal of attention from scientific researchers recently [28, 29]. Here we use Spark implementation of LDA model algorithm. The model LDA takes as input the high-quality 'bag-of-phrases', and its result is represented in one matrix, a $m \times n$ matrix, where m is the number of users and n is the number of topics. Each row vector is represented as a probability distribution over n topics that each user is interested in. As Table III shows, a sample of LDA output is displayed. The first line of data in the table indicates that the first person is interested in the fourth topic. In contrast, the second person rarely publishes the messages about these four topics.

We choose the high-quality 'bag-of-phrases' as the input of LDA over the original 'bag-of-words' to ensure that tokens in the same word are assigned to the same topic. That is not only why original words need to be merged, but also an innovation point of our work. Just because of this, the topic feature can be obtained for the following experiments.

TABLE III. A SAMPLE OF LDA OUTPUT RESULT

Topic1	*Topic2*	*Topic3*	*Topic4*	…
0.0067069	0.0053117	0.0055864	0.3058648	…
0.0085345	0.0104205	0.0099841	0.0071137	…

V. EXPERIMENTS

In order to predict user influence, a set of experiments were conducted on real world datasets from Sina Weibo. First, we describe our experimental environment and evaluation metrics that are the basis of experiments. Second, a comparison experiment is presented to demonstrate the performance of predictive models, especially after adding topic feature.

A. Experimental Environment and Evaluation Metrics

In this work, all experiments were carried out by using big data mining techniques. For this reason, Spark was built with a total of four high-performance servers. The training datasets and the testing datasets were both put on HDFS [30], which is beneficial to parallel computing. HDFS is the abbreviation of Hadoop Distributed File System, which is a popular distributed data storage platform.

To assess prediction accuracy of regression models, we use two different evaluation metrics: $RMSE$ and R^2. They are given by

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{3}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

where y_i, $\hat{y}_i$ are the actual values and predicted values respectively, and $\bar{y}$ is the mean of the actual values. Obviously, the smaller $RMSE$ is, the better the accuracy is. On the contrary, R^2 is higher, the result will be better. The former value is 0 and the latter equals 1 under the perfect condition.

B. Predicting User Influence Using Statistical Features

For the purpose of predicting individual influence, we compared different performance of three popular regression models: Decision Tree, Random Forest and Gradient Boosting. Both Random Forest and Gradient Boosting are algorithms for learning ensembles of trees, but the training processes are different [31]. Because of the need of big data analysis, we used Spark implementation of these algorithms.

In the experiments, we aggregated all seed posts of each user and computed individual level influence for each user. One's second-year influence is used as the label, and his 20 features including 13 statistical features about user attributes, 6 statistical features about the content posted in the first year and the first-year influence score were input into our models. While 80% of users were selected randomly as training datasets, the other 20% users were regarded as testing datasets. For the 20% users, we compared predicted

influence scores with actual influence scores computed from the second-year data.

By doing so, Table IV shows different performance of three models on the task of predicting individual influence. This observation shows that Gradient Boosting performs better than the other two models.

TABLE IV. THE PERFORMANCE OF THE THREE MODELS WITH ONLY 19 STATISTICAL FEATURES AND PAST INFLUENCE

Predictive models	*RMSE*	*R-squared*
Random Forest	2.5432	0.5534
Decision Tree	2.4287	0.5628
Gradient Boosting	2.3819	0.5691

C. Predicting User Influence adding Topic Feature

Conceivably, we could do better at predicting individual influence if knowing the semantic attribute of the content. Therefore, we repeated the analysis of these models after adding the topic feature.

TABLE V. THE PERFORMANCE OF THE THREE MODELS AFTER ADDING TOPIC FEATURE

Predictive models	*RMSE*	*R-squared*
Random Forest	1.9214	0.6048
Decision Tree	1.7358	0.6152
Gradient Boosting	1.5962	0.6236

It can be observed that all the metrics of these models in Table V are better than Table IV's. In other word, the three models are improved significantly by the addition of the topic feature. It is worth noting that Gradient Boosting is still the best of all the algorithms.

In summary, the topic feature is an effective feature that can extract some useful information for better predicting user influence. The individual influence of each user can be predicted with a satisfactory accuracy.

VI. CONCLUSIONS

Our work focuses on the problem of predicting individual influence for each user under the environment of big data. In this paper, a new attempt to measure user influence is proposed. User influence is defined as the potential of one's actions that motivates others to republish or reply to his messages. This definition measures the influence taking both the quantity of messages posted and their popularity into account. Thus, we can compute the influence score of each user with this criterion. At the same time, we also find that most individual influences change over time. As everyone knows, ex ante forecast is better than ex post analysis. To predict the influence of each user in the future, we analyze in detail 19 statistical features, past influence and topic feature. What's more, the phrase merging algorithm we propose improves the output quality of LDA model, which effectively extract the topic feature. Thus, three popular regression models implemented by Spark are used to successfully predict the influence score of each user, especially after adding topic information.

ACKNOWLEDGMENT

This work is partially supported by the National Natural Science Foundation of China (Nos. 11461141004, 61271426, U1536117, 11504406, 11590770-4), the Strategic Priority Research Program of the Chinese Academy of Sciences (Nos. XDA06030100, XDA06030500, XDA06040603), National 863 Program (No. 2015AA016306), National 973 Program (No. 2013CB329302) and the Key Science and Technology Project of the Xinjiang Uygur Autonomous Region (No. 201230118-3), Postdoctoral Funding (2015LH0041).

REFERENCES

[1] B. Krishnamurthy, P. Gill, and M. Arlitt, "A few chirps about twitter," in Proc. WOSN, Washington, 2008, pp. 19-24.

[2] W. Liu, K. Niu, Z. He, and Y. Li, "Trend prediction of hot words in weibo based on fuzzy time series," IEEE International Conference on Cloud Computing and Big Data Analysis, 2016, pp. 354-358.

[3] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in twitter: The Million Follower Fallacy," in Proc. ICWSM, 2010, pp. 10-17.

[4] I. Anger and C. Kittl, "Measuring influence on twitter," in Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies, Graz, Austria, 2011, pp. 1-4.

[5] P. Brown and J. Feng, "Measuring user influence on twitter using modified K-Shell decomposition," in Proc. ICWSM, Barcelona, Catalonia, Spain, 2011, pp. 18-23.

[6] A. Silva, S. Guimarães, W. Meira Jr, and M. Zaki, "ProfileRank: finding relevant content and influential users based on information diffusion," in Proc. SNAKDD, Chicago, Illinois, 2013, pp. 1-9.

[7] J. Weng, E. P. Lim, J. Jiang, and Q. He, "TwitterRank: finding topicsensitive influential twitterers," in Proc. WSDM, New York, USA, 2010, pp. 261-270.

[8] B. Lucier, J. Oren, and Y. Singer, "Influence at scale: distributed computation of complex contagion in networks," in Proc. KDD, Sydney, NSW, Australia, 2015, pp. 735-744.

[9] V. R. Embar, I. Bhattacharya, V. Pandit, and R. Vaculin, "Online topicbased social influence analysis for the wimbledon championships," in Proc. KDD, Sydney, NSW, Australia, 2015, pp. 1759-1768.

[10] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, "Influence and passivity in social media," in Proc. ECML PKDD, Athens, Greece, 2011, pp. 18-33.

[11] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in Proc. WSDM, Hong Kong, China, 2011, pp. 65-74.

[12] Y. Ye, Y. Du, and X. Fu, "Hot topic extraction based on Chinese Microblog's Features topic model," IEEE International Conference on Cloud Computing and Big Data Analysis, 2016, pp. 348-353.

[13] L. Akritidis, D. Katsaros, and P. Bozanis, "Identifying influential bloggers: time does matter," in Proc. WI-IAT, Milan, Italy, 2009, pp. 76-83.

[14] K. McDonald, "Physicist proposes new way to rank scientific output," PhysOrg, 2010.

[15] K. Lee, J. Mahmud, and J. Chen, "Who will retweet this?: automatically identifying and engaging strangers on twitter to spread information," in Proceedings of the 19th International Conference on Intelligent User Interfaces, Israel, 2014, pp. 247-256.

[16] S. Petrovic, M. Osborne, and V. Lavrenko, "RT to win! Predicting message propagation in twitter," in Proc. ICWSM, California, 2011, pp. 586-589.

[17] X. Yang, Y. Wang, and W. Qiao, "Social Network analysis on Sina Weibo based on K-means algorithm," IEEE International Conference on Cloud Computing and Big Data Analysis, 2016, pp. 127-132.

[18] T. R. Zaman, R. Herbrich, J. Van Gael, and D. Stern, "Predicting information spreading in twitter," Workshop on Computational Social Science and the Wisdom of Crowds, 2010, pp. 17599-17601.

[19] W. Webberley, S. Allen, and R. Whitaker, "Retweeting: a study of message-forwarding in twitter," Workshop on Mobile and Online Social Networks, 2011, pp. 13-18.

[20] B. Suh, L. Hong, P. Pirolli, and E. H. Chi, "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network," in Social computing (socialcom), Minneapolis, 2010, pp. 177-184.

[21] J. Yang and S. Counts, "Predicting the speed, scale, and range of information diffusion in twitter," in Proc. ICWSM, 2010, pp. 355-358.

[22] P. Bao, H. W. Shen, J. Huang, and X. Q. Cheng, "Popularity prediction in microblogging network: a case study on sina weibo," in Proceedings of the 22nd International Conference on World Wide Web, Brazil, 2013, pp. 177-178.

[23] J. Sun. (2016, Sep.) The Github website. [Online]. Available: https://github.com/NLPchina/ansj seg.

[24] A. El-Kishky, Y. Song, C. Wang, C. R. Voss, and J. Han, "Scalable topical phrase mining from text corpora," in Proc. VLDB, Kohala Coast, Hawaii, 2015, pp. 305-316.

[25] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," Journal of Machine Learning Research, Jan. 2003, pp. 993-1022.

[26] D. M. Blei, "Probabilistic topic models," Communications of the ACM, 2012, pp. 77-84.

[27] T. L. Griffiths and M. Steyvers, "Finding scientific topics," in Proceedings of the National Academy of Sciences of the United States of America, 2004, pp. 5228-5235.

[28] J. G. Shanahan and L. Dai, "Large scale distributed data science using apache spark," in Proc. KDD, Sydney, NSW, Australia, 2015, pp. 2323-2324.

[29] N. Pentreath, "Building a regression model with spark," in Machine Learning with Spark, 1nd ed., Birmingham, 2015, pp. 253-295.

[30] (2016, Sep.) The Hadoop website. [Online]. Available: http://hadoop.apache.org.

[31] (2016, Sep.) The Spark website. [Online]. Available: http://spark.apache.org/docs/latest/mllib-classification-regression.html.